

パルスニューラルネットワークにおける ネットワーク拡張型強化学習アルゴリズム

瀧田 航一朗 長名 優子 萩原 将文

慶應義塾大学 理工学部 情報工学科
〒 223-8522 横浜市港北区日吉 3-14-1
Tel: 045-563-1151 (内線 3336)
takita@soft.ics.keio.ac.jp

あらまし 本報告では、生体の神経細胞を模したパルス駆動型ニューロンによる新しい階層型ネットワーク構造と、そのためのネットワーク拡張型強化学習アルゴリズムを提案する。提案する学習アルゴリズムでは、偶発性を利用して時系列的な入出力空間を探索し、外部からの強化信号に基づいて望ましい出力を出すように学習する。ネットワークは入力層、隠れ層、出力層の三層からなり、すべてのニューロンはパルス駆動型の素子である。学習処理では結合荷重の修正に加え、入出力関係に対応した隠れ層ニューロンの動的な追加を行い、入力パルス列に対する望ましい出力を学習する。計算機シミュレーションによって、提案するアルゴリズムの学習性能とその特徴を分析した。

キーワード パルスニューラルネットワーク, 強化学習

Reinforcement learning algorithm with network extension for pulse neural network

TAKITA Koichiro, OSANA Yuko and HAGIWARA Masafumi

Department of Information and Computer Science,
Faculty of Science and Technology, Keio University
3-14-1 Hiyoshi, Kohoku-ku, Yokohama 223-8522 Japan
Tel: 045-563-1151 (ext. 3336)
takita@soft.ics.keio.ac.jp

Abstract In this report, we propose a new hierarchical neural network composed of pulse neurons which imitate real neurons and its reinforcement learning algorithm with network extension. In the proposed algorithm, sequential input-output space is investigated by making use of randomness, so the network learns desired output indicated by reinforcement signal. The proposed pulse neural network has three layers: an input layer, a hidden layer and an output layer. All of the neurons are pulse neurons. This network learns relations between input pulse sequences and the desired outputs by updating connection weights and adding neurons corresponding to the input-output relation dynamically. We carried out the computer simulation to confirm the performance of the proposed algorithm.

key words pulse neural network, reinforcement learning

1 はじめに

生物の脳は高い情報処理能力を有するが、脳の中でどのような形で情報がコーディングされ、処理されているかは未だに明らかになっていない。しかし近年、細胞の平均発火率が情報を表現しているとする単一細胞仮説 [1] や Hebb アセンブリ仮説 [2] に対し、細胞の発火のタイミングも重要な意味をもつとするテンポラルコーディング [3, 4] が提唱されてきている。またこのような見地から、時空間的な依存性を持った細胞集団が情報コーディングの基本単位であるとするダイナミカルセルアセンブリ仮説 [5] に基づいた研究も進められている。また、生理学的な実験においても、テンポラルコーディングやダイナミカルセルアセンブリ仮説を支持するような結果が報告されている [6]。

人工ニューラルネットワークの分野においても、生体の神経細胞における平均発火率の概念を元にした積分器型のニューロン素子だけでなく、近年ではパルス（スパイク）に基づいた入出力をモデル化したパルスニューロン素子が考案されている [7, 8]。パルスニューロン素子は生体の神経細胞をより詳細にモデル化したものであると言え、その導入により生体の神経細胞に見られるような高次の情報コーディングへの道が開けると期待される。

武田らは、パルス駆動型ニューロンの階層構造における学習則を提案し、時系列符号化を達成している [9]。また、塚田らは、海馬神経細胞における実験に基づき、高いパターン分離機能を持つ時空間学習則を提案している [10]。これらの手法は Hebb が提案した学習則 [2] を時間軸について拡張したものと位置づけられ、符号化問題については有効であるものの、それ以外の問題への応用を旨としたものではない。雨森らの連想記憶モデル [11]、黒柳らの音源定位モデル [12] など提案されているが、特定の問題しか扱うことができない。パルス駆動型ニューロンモデルにおいて汎用的に利用可能な学習則はいまだに確立されていないというのが現状である。

以上のような観点から、本論文ではパルス駆動型ニューロン素子を用いた新しいネットワーク構造と、そのための強化学習アルゴリズムを提案する。このモデルは偶発性を利用して入出力空間の探索を行い、強化信号に基づいた学習を実現するものである。また、時間的な相関を有すると推測される入出力に対しこれらを結ぶ隠れ層ニューロンを追加することによって、学習を達成する。

2 ネットワークモデル

2.1 ネットワーク構造

提案モデルは入力層、隠れ層、出力層の三層からなる階層構造のネットワークで、各層は 2.2 で述べるパルス駆動型ニューロン素子によって構成されている。このネットワークにおいて、ニューロンは一つ上の層のいくつかのニューロンとのみ結合しており、層間の結合は全結合ではない。また、同じ層内のニューロン間の結合は存在しない。

2.2 パルス駆動型ニューロン

提案モデルで用いたパルス駆動型ニューロン素子を図 1 に示す。このモデルでは、実際の神経細胞に見られる不応性や信号の時間的な加算などを考慮している。また、入出力としてパルス列を扱うことができ、従来の積分器型のニューロンモデルに比べ、より実際の神経細胞に近いモデルになっている。

このパルス駆動型ニューロンモデルでは、ある層のニューロン i に前階層のニューロン j からの入力パルスが到達すると、ニューロン i の内部電位 V_i は結合荷重 W_{ji} の分だけ上昇し、時間の経過とともに徐々に静止電位まで減衰していく。内部電位が閾値を越えると同時にニューロンは発火し、出力パルスが時間遅れののちに次階層に到達する。発火したニューロンの内部電位は静止電位にリセットされるとともに、不応性の影響を受け一時的にさらに電位が低下する。この不応性の影響も、時定数に則り徐々に減衰していく。またこのモデルでは、偶発的なパルス（ランダムパルス）の影響も受ける。これは個々のニューロンにおいてフラストレーション値と呼ばれるパラメータに依存して与えられるパルスであり、学習に利用される。なお、フラストレーション値については 3.2 で説明する。

ニューロン i の時刻 t における内部電位 $V_i(t)$ は、他のニューロンからの入力パルスによる影響 $P_i(t)$ 、不応性による影響 $R_i(t)$ 、フラストレーション値に依存したランダムパルスによる影響 $\lambda_i(t)$ によって、式 (1)~(4) のように定義される。

$$V_i(t) = P_i(t) + R_i(t) + \lambda_i(t) \quad (1)$$

$$P_i(t) = \begin{cases} d_v \cdot P_i(t-1) + \sum_j W_{ji}(t-k_d) \cdot O_j(t-k_d), & O_i(t-1) = 0 \\ 0, & O_i(t-1) = 1 \end{cases} \quad (2)$$

$$R_i(t) = \begin{cases} d_r \cdot R_i(t-1) - k_r, & O_i(t-1) = 1 \\ d_r \cdot R_i(t-1), & O_i(t-1) = 0 \end{cases} \quad (3)$$

$$\lambda_i(t) = \begin{cases} d_v \cdot \lambda_i(t-1) + r(F_i(t)), & O_i(t-1) = 0 \\ 0, & O_i(t-1) = 1 \end{cases} \quad (4)$$

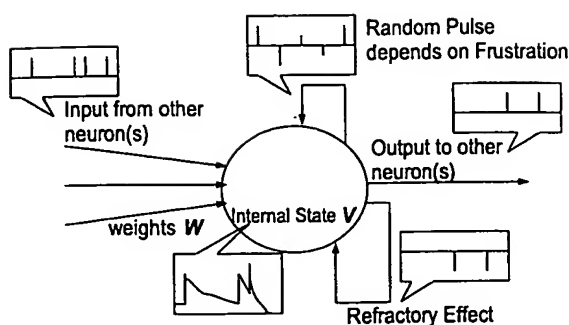


図 1: パルス駆動型ニューロン素子

ここで、 d_v は内部電位の減衰率であり、 k_d はパルス伝搬のディレイ、 $W_{ji}(t-k_d)$ はニューロン j からニューロン i への時刻 $t-k_d$ における結合荷重、 $O_j(t-k_d)$ はニューロン j の出力をそれぞれ示す。 d_r は不応性の影響の減衰率を、 k_r は一回の発火がニューロンに与える不応性の影響の大きさを示す。また、 $r(F_i(t))$ は、 $-F_i(t) \sim F_i(t)$ の範囲の一様乱数で、ランダムパルスの大きさを表す。なお、 $F_i(t)$ はニューロンのフラストレーション値を示すもので、3.2 で説明する。

ニューロン i の時刻 t における出力 $O_i(t)$ は、次式で定義される。

$$O_i(t) = \begin{cases} 1, & V_i(t) \geq \theta_v \\ 0, & V_i(t) < \theta_v \end{cases} \quad (5)$$

ここで、 θ_v はニューロンの発火の閾値を表す。

3 ネットワーク拡張型強化学習アルゴリズム

ここでは、提案するパルスニューラルネットワークにおけるネットワーク拡張型強化学習アルゴリズムについて説明する。

3.1 概要

提案アルゴリズムは、(1) ネットワーク拡張処理、(2) 結合荷重修正処理、(3) 動作安定化処理、(4) 再不安定化処理の4つの処理から構成されている。これらの処理はネットワークに対して外部から与えられる強化信号によって誘発されるが、正の強化信号を特に報奨信号と呼び、負の信号を罰信号と呼ぶこととする。

提案モデルは、2.1 でも述べたように、入力層、隠れ層、出力層の三層によって構成されるが、初期状態においては隠れ層ニューロンは存在せず、学習の進行に応じて追加されていく。提案アルゴリズムでは、ネット

ワークの出力に対して報奨信号が与えられない場合には、各ニューロンのフラストレーション値が増大していく。各ニューロンはフラストレーション値に依存して生じるランダムパルスの影響を受けて、次第に不安定な出力を出すようになる。報奨信号が与えられた場合には、フラストレーション値は大幅に減少する。なお、学習の開始時には一切の隠れ層ニューロンが存在しないため、新しいニューロンが追加されるまでは出力層ニューロンはランダムパルスの影響のみを受けることになる。

3.2 フラストレーション値

提案アルゴリズムでは、フラストレーション値に依存したランダムパルスによる偶発性を利用して学習を行う。

時刻 t におけるニューロン i のフラストレーション値 $F_i(t)$ を、次式のように定義する。

$$F_i(t) = \begin{cases} F_i(t-1) + f_i(t), & S(t) \leq 0 \text{ and } F_i(t-1) < \theta_f \\ 0, & F_i(t-1) \geq \theta_f \\ 0, & S(t) > 0 \text{ and } |R_i(t)| \geq |\theta_r| \\ D(t) \cdot F_i(t-1), & S(t) > 0 \text{ and } |R_i(t)| < |\theta_r| \end{cases} \quad (6)$$

ここで、 $f_i(t)$ は時刻 t におけるフラストレーション値の増加量を表す。隠れ層ニューロンおよび出力層ニューロンについては $f_i(t)$ の初期値は正の微量 k_f であり、入力層ニューロンについては $f_i(t)$ の初期値は0である。また $D(t)$ は、時刻 t における、フラストレーション値の解消を調整する変数であり、 D_{init} とする。また $S(t)$ は時刻 t における強化信号、 θ_f はフラストレーション値に関する閾値、 θ_r は不応性に関する閾値を示す。 $R_i(t)$ は時刻 t におけるニューロン i の不応性の影響で、式(3)で与えられる。

3.3 ネットワーク拡張処理

ネットワーク拡張処理は、報奨信号を利用して新しい入力列に対応した隠れ層ニューロンを追加する処理である。この処理は、報奨信号が与えられた際に、その報奨信号と因果関係があると推定される入力層ニューロン全てと出力層ニューロンとを繋げるような結合を持つ隠れ層ニューロンが存在しない場合に行われる。

3.3.1 強化信号とニューロンの因果関係

強化信号(報奨ないし罰信号)とニューロンとの間に因果関係があるかどうかを判別する基準として、提案アルゴリズムでは、ニューロンに残存する不応性 $R_i(t)$ に着目する。強化信号が与えられた際に不応性の影響が閾値を越えて残っている、つまり

$$|R_i(t)| \geq |\theta_r| \quad (7)$$

であるようなニューロンは最近発火したと考えられ、強化信号に何らかの関係があると推測される。

3.3.2 実行条件

ネットワーク拡張処理は、出力層ニューロン k について以下の条件が成り立つ時に実行される。

1. 時刻 t において報奨信号が与えられている。すなわち、

$$S(t) > 0 \quad (8)$$

が成り立つ。

2. 不応性の影響 $R_k^O(t)$ が閾値 θ_r よりも大きい、すなわち、

$$|R_k^O(t)| \geq |\theta_r| \quad (9)$$

が成り立つ。

3. 入力層ニューロンのいずれかについて

$$|R_i^I(t)| \geq |\theta_r| \quad (10)$$

が成り立つ。

4. 出力層ニューロン k と結合する隠れ層ニューロン j の中に、

$$|R_j^H(t)| \geq |\theta_r| \quad (11)$$

が成り立ち、かつ隠れニューロン j と結合する全ての入力層ニューロンについて式 (10) が成り立つようなものが存在していない。

3.3.3 隠れ層ニューロンの追加

3.3.2 で述べた条件が全て満たされた場合には、新たに隠れ層ニューロンが追加される。新しいニューロンは、式 (10) の成り立つ入力層ニューロン全てと、式 (9) が成り立つ出力層ニューロンとの間に結合を持つ。

新たに追加する隠れ層ニューロンを m とすると、入力層ニューロン i から隠れ層ニューロン m への結合 W_{im}^{IH} は

$$W_{im}^{IH} = W_{init}^{IH} \quad (12)$$

と設定される。ここで、 W_{init}^{IH} は正の微小量である。また、隠れ層ニューロン m から出力層ニューロン k への結合は

$$W_{mk}^{HO} = \theta_v \quad (13)$$

とする。ここで θ_v はニューロンの発火の閾値である。

3.4 結合荷重修正処理

結合荷重修正処理は、報奨信号を利用して、結合荷重を強化する処理である。報奨信号が与えられる前に発火したニューロンは、報奨信号に対し何らかの寄与があると推測される。そのとき、もし発火がランダムパルスの影響によって起きたのであれば、今後も同じ状況で安定して発火するためには、そのニューロンに対する入力を増やす必要がある。そのために、同時期に発火していた他のニューロンからの結合荷重を増大させる。

3.4.1 実行条件

結合荷重修正処理は、隠れ層ニューロン j について以下の条件が成り立つ時に実行される。

1. 時刻 t において報奨信号が与えられている。すなわち、式 (8) が成り立つ。
2. 不応性の影響 $R_j^H(t)$ が閾値 θ_r よりも大きい。すなわち、式 (11) が成り立つ。
3. 隠れ層ニューロン j と結合した出力層ニューロンを k としたとき、式 (9) が成り立つ。
4. 隠れ層ニューロン j について

$$B_j(t) = 1 \quad (14)$$

が成り立つ。

ここで、 $B_j(t)$ は、ニューロン j の発火原因を表す変数であり、

$$B_j(t) = \begin{cases} 1, & V_j(t-1) - \lambda_j(t-1) < \theta_v \\ & \text{and } V_j(t-1) \geq \theta_v \\ 0, & V_j(t-1) - \lambda_j(t-1) \geq \theta_v \\ & \text{and } V_j(t-1) \geq \theta_v \\ B_j(t-1), & \text{otherwise} \end{cases} \quad (15)$$

で与えられる。ニューロン j が最後に発火したのがランダムパルスの影響によるものであれば $B_j(t) = 1$ となり、そうでなければ $B_j(t) = 0$ となる。

3.4.2 結合荷重の修正

3.4.1 で述べた条件が全て満たされた場合には、結合荷重の修正が行われる。隠れ層ニューロン j と結合した入力層ニューロンを i からの結合荷重 W_{ij} は以下のように更新される。

$$W_{ij}(t) = \begin{cases} W_{ij}(t-1) + k_l \cdot S(t) \cdot (W_{\max} - W_{ij}(t-1))^2, & |R_j(t)| \geq |\theta_r| \\ W_{ij}(t-1), & |R_j(t)| < |\theta_r| \end{cases} \quad (16)$$

ここで、 k_l は学習係数を、 W_{\max} は結合荷重の上限値を示す。

3.5 動作安定化処理

動作安定化処理は、報奨信号を利用して、結合荷重と出力の両方を安定させる処理である。報奨信号が与えられた際に、すでに適切な結合荷重に達していると推定されるニューロン、および結合荷重を修正しても報奨信号に寄与しないと推定されるニューロンに対して適用され、以後のフラストレーション値の上昇を抑える。

3.5.1 実行条件

動作安定化処理は、隠れ層ニューロン j について以下の条件が成り立つ時に実行される。

1. 時刻 t において報奨信号が与えられている。すなわち、式 (8) が成り立つ。
2. 不応性の影響 $R_j^H(t)$ が閾値 θ_r よりも大きい、すなわち、式 (11) が成り立つ。
3. 隠れ層ニューロン j が最後に発火したのはランダムパルスの影響によらない、すなわち

$$B_j(t) = 0 \quad (17)$$

が成り立つ。

3.5.2 学習が完了しているニューロンの動作安定化

3.5.1 で述べた条件を全て満たすような隠れ層ニューロン j が存在する場合には、そのニューロンは適切な結合荷重に達していると推定され、フラストレーション値の増加量 $f_j(t)$ を次式に従って更新する。

$$f_j(t) = k_{f_1}^- \cdot f_j(t-1) \quad (18)$$

ここで、 $k_{f_1}^- (0 < k_{f_1}^- < 1)$ はフラストレーション値の増加量の減衰率を示す。

また、ある隠れ層ニューロン j についてこの処理を適用した場合には、その隠れ層ニューロンに結合している出力層ニューロンに対しても、式 (18) を適用する。

3.5.3 報奨に寄与しないニューロンの動作安定化

3.5.1 で述べた条件を全て満たすような隠れ層ニューロン j が存在し、かつ $j \neq m$ なる隠れ層ニューロン m について以下の条件が全て成り立つならば、隠れ層ニューロン m は以降どのように結合荷重を修正していても報奨に寄与しないと推定される。

1. 隠れ層ニューロン m について、

$$|R_m^H(t)| < |\theta_r| \quad (19)$$

が成り立つ。

2. 隠れ層ニューロン m と結合している全ての入力層ニューロン i について、

$$|R_i^L(t)| \geq |\theta_r| \quad (20)$$

が成り立つ。

この場合には、隠れ層ニューロン m のフラストレーション値の増加量 $f_m(t)$ を次式に従って更新する。

$$f_m(t) = k_{f_2}^- \cdot f_m(t-1) \quad (21)$$

ここで、 $k_{f_2}^- (0 < k_{f_2}^- < 1)$ はフラストレーション値の増加量の減衰率を示す。

3.5.4 パラメータ $D(t)$ の更新

θ_r 回の報奨信号に対して連続して動作安定化処理が実行された場合、フラストレーションの解消を調整するパラメータ $D(t)$ を次式に従って更新する。

$$D(t) = D(t-1) + k_d \quad (22)$$

ここで、 k_d は正の定数値であり、 $D(t)$ の増加量を示す。

報奨信号が安定して与えられるようになると、式 (6) に従い、各ニューロンのフラストレーション値は低く抑えられ、それ以降の学習が難しくなる。そこで、安定した報奨信号が連続して与えられている場合には $D(t)$ を増加させ、学習が完了していない隠れ層ニューロンの学習を促す。

3.6 再不安定化処理

再不安定化処理は、罰信号を利用して、まだ完全には学習が済んでいないニューロンに学習を促す処理である。大部分の状況には正しく反応するが、それ以外のいくつかの状況に対応できていないニューロンの動作が安定化してしまった場合には、この処理によって再不安定化させる。

3.6.1 実行条件

再不安定化処理は、隠れ層ニューロン j について以下の条件が成り立つ時に実行される。

1. 時刻 t において罰信号が与えられている。すなわち、

$$S(t) < 0 \quad (23)$$

が成り立つ。

2. 不応性の影響 $R_j^H(t)$ が閾値 θ_r よりも大きい。すなわち、式 (11) が成り立つ。
3. 隠れ層ニューロン j が最後に発火したのはランダムパルスの影響によらない。すなわち式 (17) が成り立つ。

3.6.2 再不安定化

3.6.1 で述べた条件が全て満たされた場合には、隠れ層ニューロン j の学習はまだ完了していないものと推定され、フラストレーション値の増加量 $f_j(t)$ を次式に従って更新する。

$$f_j(t) = k_f^+ \cdot f_j(t-1) + (1 - k_f^+) \cdot k_f \quad (24)$$

ここで k_f は $f_j(t)$ の初期値であり、 k_f^+ は $f_j(t)$ の増加を制御する定数値である。

4 計算機シミュレーション

4.1 テニスゲーム環境

このシミュレーションでは、提案モデルを使って、ごく簡単なテニスゲームを実行した。これは、図 2 に示すように、横 4 マス縦 6 マスの領域のなかで、ラケットを左右に移動させながらボールを打ち返すゲームで、提案モデルを用いてラケットを操作し、ボールを落とさないようにラリーを続けるのが目的である。

ボールは、側面の壁かラケットに当たると跳ね返る。領域上部には相手のプレーヤーがいるものと考え、ここにボールが到達しても必ず跳ね返るものとする。もしボールが領域から下へ出てしまった場合には、ランダムな時間の後に画面上部のランダムな位置から、 $(-1, 1)$ ないし $(1, 1)$ の運動ベクトルを持った新たなボールが投じられる。ただし、これではラケットを僅かに動かすだけで安定したラリーが保たれてしまうので、ボールが領域上部で跳ね返る場合にはその x 座標をランダムに変更するものとした。

提案モデルへの入出力は、次のようにした。4×6 の領域のうち、最下段を除く 20 のマスに対応して 20 の入力層ニューロンを用意し、一定時間ごとに、ボールが存在する位置に対応したニューロンに、発火閾値と等しい大きさの入力パルスを与える。この時間幅は、ネットワークの単位時間 (以下、ステップと呼ぶ) にして 12 ステップであり、これを 1 サイクルと呼ぶ。なお、ボールの移動も 1 サイクルごとに (斜め方向に) 1 マスである。

また、出力層ニューロンは、4 マスの横幅に対応させて 4 つ用意した。これらはラケットの目標位置を示すもので、いずれかが発火すると、ラケットは対応する位置に移動する。ただし、1 サイクルに移動できるのは 1 マスのみである。移動している途中で別の出力層ニューロンが発火すると、新しい目標位置に向かって移動し始める。1 サイクルの間に複数の出力層ニューロンが発火した場合には、移動しないものとする。

ネットワークに対する報奨信号は、ラケットがボールを打ち返した瞬間に与えられるものとし、その値は

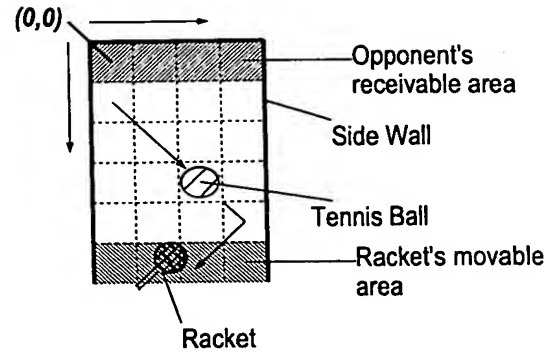


図 2: テニスゲーム環境

表 1: テニスゲームにおける定数値の設定

発火閾値	θ_v	1.0
フラストレーション値上限	θ_f	0.2
不応性閾値	θ_r	-0.01
連続報奨回数閾値	θ_s	100
パルス減衰率	d_v	0.94
不応性減衰率	d_r	0.94
不応性強度	k_r	1.0
ニューロン間ディレイ	k_d	3
フラストレーション値増加量初期値	k_f	0.01
$D(t)$ 初期値	D_{init}	0.5
学習係数	k_l	0.01
結合荷重上限値	W_{max}	1.1
f_i 減衰率 A	k_{f1}^-	0.95
f_i 減衰率 B	k_{f2}^-	0.5
$D(t)$ 増加量	k_d^+	0.01
f_i 増加率	k_f^+	0.5

1.0 とした。また、罰信号はボールを打ち返し損なった場合に与えられるものとし、その値は -1.0 とした。

具体的なパラメータは、表 1 の通りである。

4.2 学習成功率

図 3 は、4.1 において 100 回の試行を行い、6,000,000 ステップまでに学習できたパターン数の分布を表したものである。ここで、パターン数とは、ボールの軌道の種類と、そのボールが投じられた時にラケットが取りうる座標の種類とを掛けたもので、このシミュレーションではボールの軌道の種類が 6 でラケットの座標の種類が 4 であるから、パターン数は 24 である。図 3 より、全ての状況に対して正しく応答できるように学習できなかった場合でも、多くのパターンに対して正しく学習できていることが分かる。

また、学習の成功率を以下のように定義すると、図 3

の場合には、93.2%となる。

$$\text{学習成功率} = \frac{\sum \text{学習できたパターン数}}{\text{学習すべきパターン数} \times \text{試行回数}} \quad (25)$$

4.3 学習完了に要するステップ数

4.2の試行において、24種類のパターン全てについて正しく学習できたものに関して、学習が完了するまでに要したステップ数の分布を図4に示す。図4より、学習が完了するまでに要するステップ数は試行によりかなりばらつきがあることが分かる。図4において、学習が完了するまでに要したステップ数の平均は2,945,357であった。

4.4 行われる処理の変化

図5は、4種類のパターン全てについて正しく学習できた典型的な試行において、30,000ステップの間に結合荷重修正処理と動作安定化処理が行われた回数が時間の経過と共にどのように変化していくかを示したものである。これを見ると、学習の初期では結合荷重修正処理が多く行われているが、学習が進んでいくにつれてその回数は減り、かわりに動作安定化処理が多く行われるようになってくることが分かる。図5では、学習開始からおよそ1,000,000ステップ付近で、結合荷重修正処理が行われた回数が激減していくのと同時に動作安定化処理が行われる回数が増えている。またその後、結合荷重修正処理がほとんど行われなくなるが、2,000,000ステップ付近で再び結合荷重修正処理が多数実行され、この時点で学習が完全に収束したのが分かる。

4.5 フラストレーション値増加量の初期値の影響

フラストレーション値増加量の初期値 k_f を0.02として100回の試行を行った。6,000,000ステップまでに学習できたパターン数の分布を図6に示す。また、正しく学習できたものに関して、学習が完了するまでに要したステップ数の分布を図7に示す。 k_f を0.01とした場合に比べ学習の収束は格段に早くなっているものの、学習の成功率はやや悪い結果となっていることが分かる。

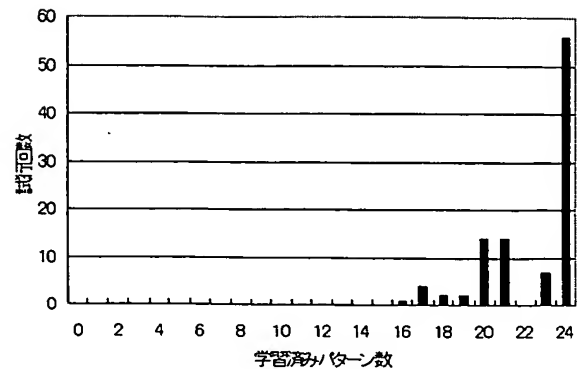


図3: 学習パターン数の分布

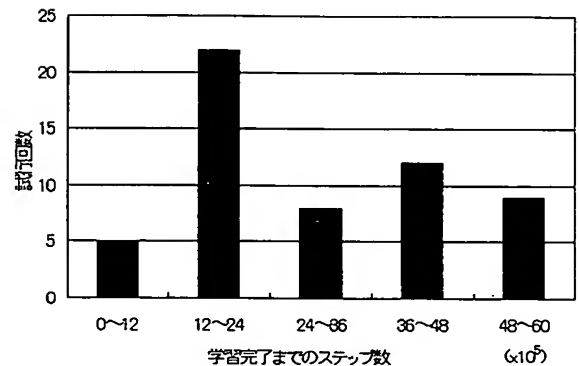


図4: 学習完了までのステップ数の分布

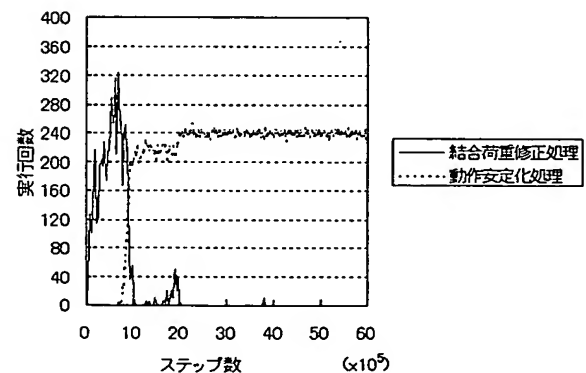


図5: 各処理の実行回数の変遷

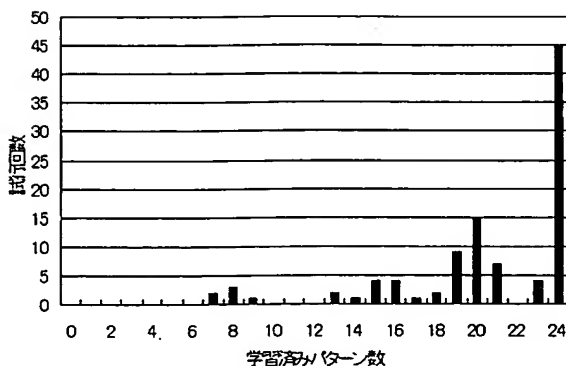


図 6: $k_f = 0.02$ とした場合の学習パターン数の分布

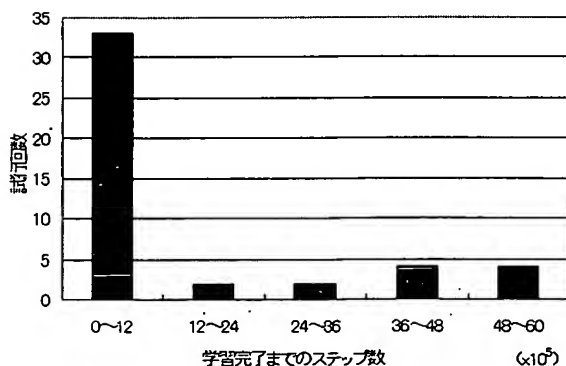


図 7: $k_f = 0.02$ とした場合の学習完了までのステップ数の分布

5 むすび

本論文では、パルス駆動型ニューロン素子を用いた新しい階層型のネットワークの構造と、そのための強化学習アルゴリズムを提案した。このモデルでは、偶発性を利用することによって時系列的な入出力空間を探索し、外部からの強化信号に基づいて望ましい出力を出すように学習を行う。計算機シミュレーションを行い、時系列的な入力に対し適切なタイミングでの出力を学習できることを確認した。

提案モデルの今後の課題としては、学習精度の向上が挙げられる。特に、強化信号は常に正しい出力に対してのみ与えられるとは限らない。そのため、学習の過程で望ましくない出力が学習されてしまった場合にも、これを修正するような学習を行うことが必要になってくると考えられる。

参考文献

- [1] H.B.Barlow, "Single units and sensatioperceptual psychology?," Perception, Vol.1, pp.371-394, 1972.
- [2] D.O.Hebb, "The Organization of Behavior," New York: John Wiley, 1949.
- [3] A.K.Engel, P.Konig, A.K.Kreiter, T.B.Schillen and W.Singer, "Temporal coding in the visual cortex: new vistas on integration in the nervous system," Trends In Neuroscience, Vol.15, No.6, pp.218-226, 1992.
- [4] W.Singer, C.Gray, A.Engel, P.Konig, A.Artola and S.Brocher, "Formation of cortical cell assemblies," Cold Spring Harbor Symposium on Quantitative Biology, Vol. LV., Cold Spring Harbor Laboratory Press, pp.939-952, 1990
- [5] H.Fujii, H.Ito, K.Aihara, N.Ichinose and M.Tsukada, "Dynamical Cell Assembly Hypothesis - Theoretical Possibility of Spatio-temporal Codeing in the Cortex," Neural Networks, Vol.9, No.8, pp.1303-1350, 1996.
- [6] 櫻井芳雄, "ニューロン集団の相関活動をみる," 科学, Vol.66, no.11, pp.784-792, 1996.
- [7] 市瀬夏洋, 合原一幸, "非同期カオスニューラルネットワークにおけるパルス伝搬ダイナミクスの解析," 電子情報通信学会誌 A, J58-A, pp.373-380, 1995.
- [8] W.Maass and C.M.Bishop, "Pulsed Neural Networks," Cambridge, MA: M.I.T. Press, 1999.
- [9] 武田治, 黒柳奨, 岩田彰, "階層構造を持つパルス駆動型ニューロンモデルを用いた時系列符号化," 信学技報, NC97-117, pp.125-132, 1998.
- [10] 塚田稔, "海馬記憶神経回路と学習則," 日本神経回路学会誌, Vol.4, No.3, pp.126-135, 1997.
- [11] 雨森賢一, 石井信, "スパイク依存のヘブ型学習則による精緻な時空間パターンの連想記憶," 信学技報, NC98-111, pp.99-114, 1999.
- [12] 黒柳奨, 岩田彰, "パルスニューロンモデルのための教師あり学習則," 信学技報, NC97-151, pp.95-102, 1998.